

# The Phoenix Project

A DevOps  
business simulation  
by GamingWorks



## About AutoTrader and their DevOps journey

*AutoTrader Group plc is the UK and Ireland's largest digital automotive marketplace. Over 80% of UK automotive retailers advertise on [autotrader.co.uk](https://www.autotrader.co.uk) and around 80% of all used cars sold through the motor trade appear on the site. It attracts circa 48 million cross platform visits each month, with circa 70% of visits coming through mobile devices. More than 80% of all time spent on UK automotive classified sites is spent on AutoTrader. AutoTrader listed on the London Stock Exchange in March 2015 and is now a member of the FTSE 250 Index.*

AutoTrader





## Devops journey

7 years ago at Auto Trader Dev and Ops were 2 separate entities both structured in a very traditional way. Ops were very protective of the live environment and had quite a few controls and a bureaucratic Change Management process in place. The Dev teams then began to adopt Agile working practices. As their continuous delivery capabilities grew the Ops teams had to amend how they accommodated the increased flow of smaller releases to be implemented into the live environment. They did this by building up relationships with the dev teams and streamlining processes to make it easier for the developers to submit their code for release.

They set up a Continuous Delivery Squad bringing together a cross-functional team of Ops and Development specialists to focus on automating and improving their monitoring and release capability.

This automation and process improvement, in turn, meant that Ops could further relax the controls they had in place to protect the live environment. This gave product squads the ability to release their own code into the live environment whenever they needed to and as often as they liked and encouraged more operational responsibility within product teams. At the same time, Auto Trader Product and Dev teams merged (they adopted the Spotify organizational model) with small product focused squads each managed by a Product Lead, a Tech Lead and a Delivery Lead. This improved cross functional collaboration which led to faster, better product development.

Just after the structure change the squads and Ops were co-located in a single floor in the same building. Prior to this they

were split across a number of buildings and floors which acted as a barrier to good collaboration and communication. The Development teams then moved to continuous integration with releases becoming much smaller and more frequent – over 50 per week. 18 months after the Product and Dev teams merged, Ops restructured to form cross-functional squads (rather

*“This simulation shows the real interaction within Dev and Ops teams dealing with business projects and issues.”*

than separate Sys Admin, Network, DBA teams etc) and really began to work in an Agile way similar to the Product and Dev squads. The aim is to deliver a private cloud PaaS environment.



This has meant incorporating Dev resource as an integral part of the Ops squads, and building pipelines to release Ops code etc.

The Product and Dev squads now take responsibility for how their products function and perform. Ops now take responsibility for the underlying platform rather than acting as gate-keepers of the live environment. If a problem does occur where it isn't clear if it is a product or environmental issue there is close collaboration between the 2 to ascertain the root cause and resolve it. The Dev squads now control the schedule of the live releases since if a problem occurs it can usually be forward fixed or backed out very quickly and easily. The continuous integration and automated testing that is carried out with

small, frequent releases mean that Ops have confidence that by the time code is released into live there are likely to be minimal issues.

**“** *We are 7 years ahead. We have been through all the aspects in this simulation before we reached where we are now.*

*Having this simulation 7 years ago we probably would have made bigger steps in our journey.*

**Jonathan Leckey,**  
**Head of Operations at AutoTrader** **”**

It seems to be working as the release success rate for this year so far is 98.22% with an average of 75 releases per week and the vast majority of them auto deployed into live.



## The Simulation

### New Work

At the start of the day the VP IT Operations got his team together to explore the 'New Projects' of the day. Retail Operations, owner of the Phoenix Program, informs the team about the importance of the program to bring 'Parts Unlimited' back on track. The company is in financial difficulties, customers are not happy and the share price is dropping.

### Stand up meetings

The team investigates all the work that needs to be done during a stand-up meeting and they write them down on the flip chart. It's not clear for everybody what to do and how to do it, processes are not clear and it looks a bit ad-hoc.

### Planning Work in Progress

After they investigated all the work the team begins to plan the activities in their Work In Progress capacity. Then suddenly, the CISO comes into room "We have login errors on our website, we have serious problems!!!", followed by the CFO who received feedback from some of the financial teams that they cannot find supplier data. The atmosphere becomes a bit more hectic but after a few minutes the team agrees on the work that needs to be done to finish all the different types of work.

**“Acting as Retail Operations in this simulation, I learned a lot about DevOps. I now see how Kanban is used, what my role as the business should be. Very useful.**

**Katrina Neary, People Operations Analyst at AutoTrader**



## Lessons Learned round 1

During reflection the following lessons learned were documented:

- » We need to give all team members enough time to get used to their roles and processes.
- » We have to use visualization tools to support collaboration. The flip-chart was not enough to get a clear picture of the work.
- » We need to create a clear flow so we all know how the different types of work will move through the teams.
- » We did not share the business strategy. Some had heard about the Phoenix Program, but not everyone was aware of this. The Business should share this, or we should ask.

» The Business was complaining about lack of status updates. They did not know which projects were ready, which new applications would be available and if there were any issues.

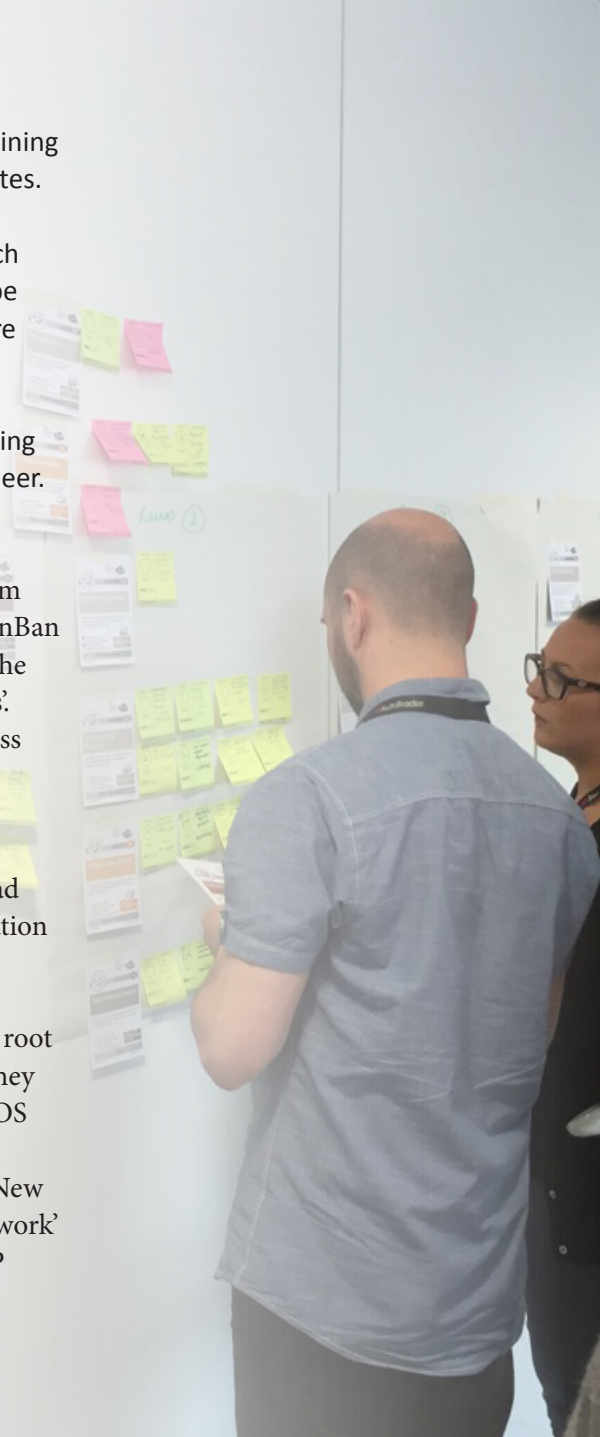
» We discovered an upcoming issue with the Lead Engineer.

### **Continous Learning**

After this reflection, the team started to create a better KanBan board and a first design of the flow 'how to solve Incidents'. They also shared the business projects and the priorities.

### **Rework**

There was however some bad news from the Retail Operation team. The POS Wireless Scanning did not work! IT Support investigated the root cause and discovered that they forgot to upgrade the POS OS version to support this new feature as a result the first 'New work' on the board was 'Rework' which took away some WIP capacity.





## **Projects, Features, Issues, Rework**

At the start of this round the team received loads of business projects and requests. The team was happy to have this new KanBan board and started to visualize all the work. IT was very happy that the business was involved and that they could set the right business priorities. They really act like a ‘Product Owner’.

## **Visualization helped**

By using this visualization tool, there was a clear picture of all the work that needed to be done in the upcoming rounds. As a result each team could work on ‘planning’ the work and could even plan some future work.

## **Disruptions, Unplanned work**

Unfortunately, there were some other types of work that came across. While the team was planning all the work using their new flow, the CFO ran into the office with bad news:

“The Payroll Run is Failing!!”

We cannot pay all our employees and the unions are not happy!”

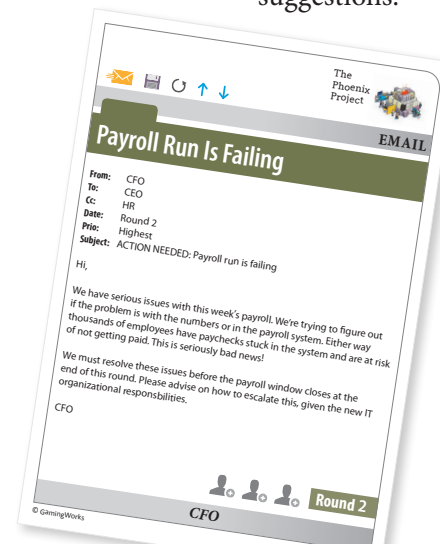
The team did not panic. They simply started a quick stand-up, discussed the situation with some of the teams and found the right actions.

The only problems was that they needed to reprioritize the planned work since there was no capacity left on the WIP.

## **Lessons learned round 2**

After round 2, the team came with the following lessons learned and improvement suggestions:

- » Kanban really helped, but we also need to visualize the WIP capacity of each team.
- » We should leave some space empty for unplanned work.
- » Sharing knowledge in the team helped us to move some issues to other teams so we could deploy faster.
- » We only defined the flow for incidents (unplanned work), we also should design flows for projects, features and IT projects.
- » We need better testing. In this round we were faster but we also made more mistakes which leads to rework.





## Business still not happy!

The rest of the day the teams received more business projects. They improved their flows and implemented shorter cycles for testing so they could deploy faster.

However, the CFO was still not happy about his share price. At this time it had to be \$25, but was stuck at \$21.75. Retail Operations expected a revenue of \$130,000 but the graphs showed only \$105,000. The CEO had some tough discussions with the VP IT Operations, HR and Retail Operations to put the pressure on both teams to solve issues faster and deliver errorless solutions.

In the next 2 rounds the team showed the real value of the DevOps principles. They improved their performance rapidly. Deployment went faster and with less errors. As a result, sales grow and the share price went up!

## Further lessons

1. Short stand-up meetings with the right people, will solve issues faster and better.
2. Shorter testing cycles and quick feedback to the teams will avoid errors and rework.
3. Separate flows for projects and unplanned work.
4. Prepare WIP capacity (90/10) to be able to handle the unplanned work faster.
5. Avoid disruptions.

## End result

The teams did not quite realise the business goals. The share price was expected to be higher as well as the revenue. But this proves that the best teams will have a tough job finishing the simulation with the expected result.

